PERGAMON

Journal of The Franklin Institute 336B (1999) 19–32

Journal
*of* The
Franklin Institute

# Fault tolerance in autonomous acoustic arrays

Warren A. Rosen[a,]*, Alan D. George[b]

[a] *ECE Department, Drexel University, Philadelphia, PA 19104, USA;* [b] *ECE Department, University of Florida, Gainesville, FL 32611-6200, USA*

Received in final form 1 July 1997; accepted 17 December 1997

**Abstract**

The problem of fault tolerance in autonomous disposable fiber-optic-based acoustic arrays is considered. The principal source of failures over relatively short mission times is node outage due to battery run-down resulting in possible network failure, degradation in the beam power pattern, and possible loss of critical processing elements. Network integrity in the presence of node failures requires an optical bypass capable of bypassing several adjacent failed nodes. The effect of node failure on the beam power pattern is principally in the side lobes rather than in the main beam, and is amenable to relatively simple solutions for the case of failures near the ends of the array, but failures near the center are more intractable. The loss of critical processing elements can be dealt with by distributing the processing load over processing elements located in each telemetry node of the network, thereby turning the array into a distributed parallel computer. © 1998 The Franklin Institute. Published by Elsevier Science Ltd

## I. Introduction

To meet the threats of quiet submarines and higher clutter in the antisubmarine warfare environment the US Navy is developing high-gain acoustic sensor arrays consisting of a large number of sensors interconnected by a fiber-optic telemetry network. These arrays are intended to be disposable and completely autonomous over mission times as long as 30 days. They will be deployed by unmanned glider or submarine and are designed to perform a large degree of in-array processing of the acoustic data and to be battery-powered [1].

Fault tolerance is a critical issue in these arrays due to the autonomous nature of the mission. At the same time, low-power operation over long mission times in a disposable system requires highly efficient techniques for detecting and correcting inevitable failures when they occur. Over mission times of weeks to months the most

* Corresponding author.

0016–0032/98/$ - see front matter © 1998 The Franklin Institute. Published by Elsevier Science Ltd. All rights reserved

PII: S0016–0032(98)00011–8

# Report Documentation Page

*Form Approved*
*OMB No. 0704-0188*

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

| | | |
|---|---|---|
| 1. REPORT DATE<br>**17 DEC 1997** | 2. REPORT TYPE | 3. DATES COVERED<br>**00-00-1997 to 00-00-1997** |

| | |
|---|---|
| 4. TITLE AND SUBTITLE<br>**Fault tolerance in autonomous acoustic arrays** | 5a. CONTRACT NUMBER<br>5b. GRANT NUMBER<br>5c. PROGRAM ELEMENT NUMBER |
| 6. AUTHOR(S) | 5d. PROJECT NUMBER<br>5e. TASK NUMBER<br>5f. WORK UNIT NUMBER |
| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)<br>**University of Florida,Department of Electrical and Computer Engineering,Gainesville,FL,32611** | 8. PERFORMING ORGANIZATION REPORT NUMBER |
| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) | 10. SPONSOR/MONITOR’S ACRONYM(S)<br>11. SPONSOR/MONITOR’S REPORT NUMBER(S) |

12. DISTRIBUTION/AVAILABILITY STATEMENT
**Approved for public release; distribution unlimited**

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON |
|---|---|---|---|---|---|
| a. REPORT | b. ABSTRACT | c. THIS PAGE | | | |
| **unclassified** | **unclassified** | **unclassified** | | **14** | |

likely cause of system failure is node outage due to battery run-down. There are three effects resulting from the loss of a node. First, a failed node may bring down the telemetry network since in a typical network topology each node is responsible for passing the entire data stream through it. Second, the lost data will result in a degraded beam power pattern. Third, if the failed node contains a critical processing element such as a standalone signal processor its loss will result in system failure.

In this paper we discuss the effects of component failure on overall system performance and describe some simple and cost-effective techniques which allow graceful degradation as these failures occur. In the following section we describe three network topologies and discuss fault tolerance issues associated with each. Next, we discuss the effects of node outage on beam power pattern and describe some simple self-healing techniques for the case in which the failure occurs near the end of the array. Finally, we describe a novel array architecture in which parallel and distributed processing techniques and algorithms together with the high bandwidth and low latency of the fiber-optic telemetry network are used to formulate architectures in which none of the processing elements are critical.

## 2. Network topology

The three network topologies under consideration are the uni-directional linear array, the uni-directional ring array, and the bi-directional linear array, shown in Fig. 1. Each is comprised of a number of nodes connected to their nearest neighbor by a point-to-point fiber-optic link. For added robustness each node contains an optical bypass switch so that it may be bypassed in the event of node failure. In a typical application an array is often embedded as a subarray in a larger array of many nodes with each subarray cut to a different frequency. Thus failures in one subarray may affect the performance of other subarrays.

Each topology has associated with it a number of advantages and disadvantages in terms of fault tolerance, cost, and support for parallel and distributed processing. The uni-directional array topology minimizes system requirements in terms of hardware and cabling, but its limitations are strongly felt in terms of potential performance increases with distributed, parallel processing. By contrast, the ring topology provides full connectivity between nodes and thus supports a variety of parallel processing algorithms. The bi-directional linear array topology takes the ring one step further by providing communication in both directions. For an array of $N$ nodes, network diameter (i.e. the maximum shortest path between any two nodes) is $N-1$ hops for the ring but only $N/2$ for the bi-directional array. Therefore, average node-to-node latency is reduced by a factor of two. Of course these performance increases come with an increase in cost, complexity, and power consumption, as the ring requires double the cabling of the uni-directional array, and the bi-directional array requires double the networking hardware of the ring.

Reliability is a critical concern with all three topologies in terms of failed nodes or link interfaces. While an individual failed node may be bypassed, the number of adjacent nodes which may be bypassed without causing the link to fail is limited by

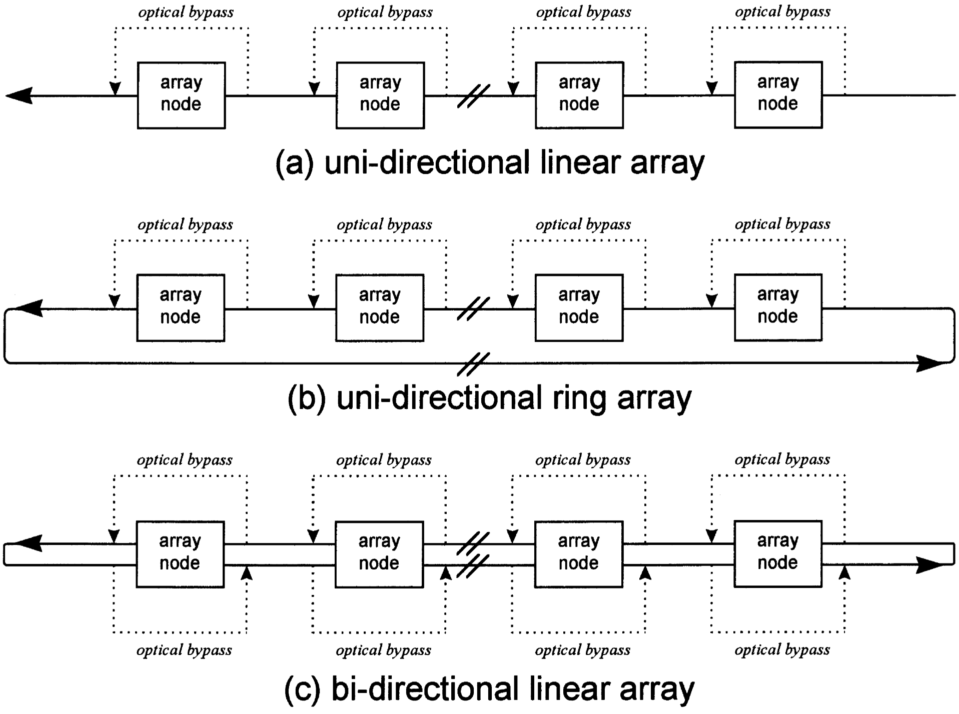
optical bypass
array node
(a) uni-directional linear array
(b) uni-directional ring array
(c) bi-directional linear array

Fig. 1. Network architectures.

the accumulated insertion loss of the bypass switches. Should a link fail in the uni-directional array, a new subarray is formed which consists of all nodes downstream of the failed node. The best case scenario is the failure of the upstream-most node leaving $N-1$ nodes in the subarray, the worst case is the failure of the downstream-most node leaving no nodes operable, and the average case leaves approximately half the nodes operable. Should a link fail in the ring array, the system can resort to the behavior of the uni-directional array in terms of subarray reconfiguration. The extra hardware associated with the bi-directional array architecture does permit certain failures to be less catastrophic (e.g. reconfiguration after the failure of a link interface in the upstream direction would still permit the system to take on an $N$-node uni-directional behavior), but the reliability of the set of all components in the system is decreased due to the increase in the number of units in the set. Furthermore, the bi-directional array topology allows the network to "cut and paste" around many failures, resulting in fully-functional subarrays of reduced size.

## 3. Beam power pattern effects

The biggest effect of node failure on the beam power pattern is in the side lobe pattern rather than in the main beam. Figure 2 shows the beam power pattern for

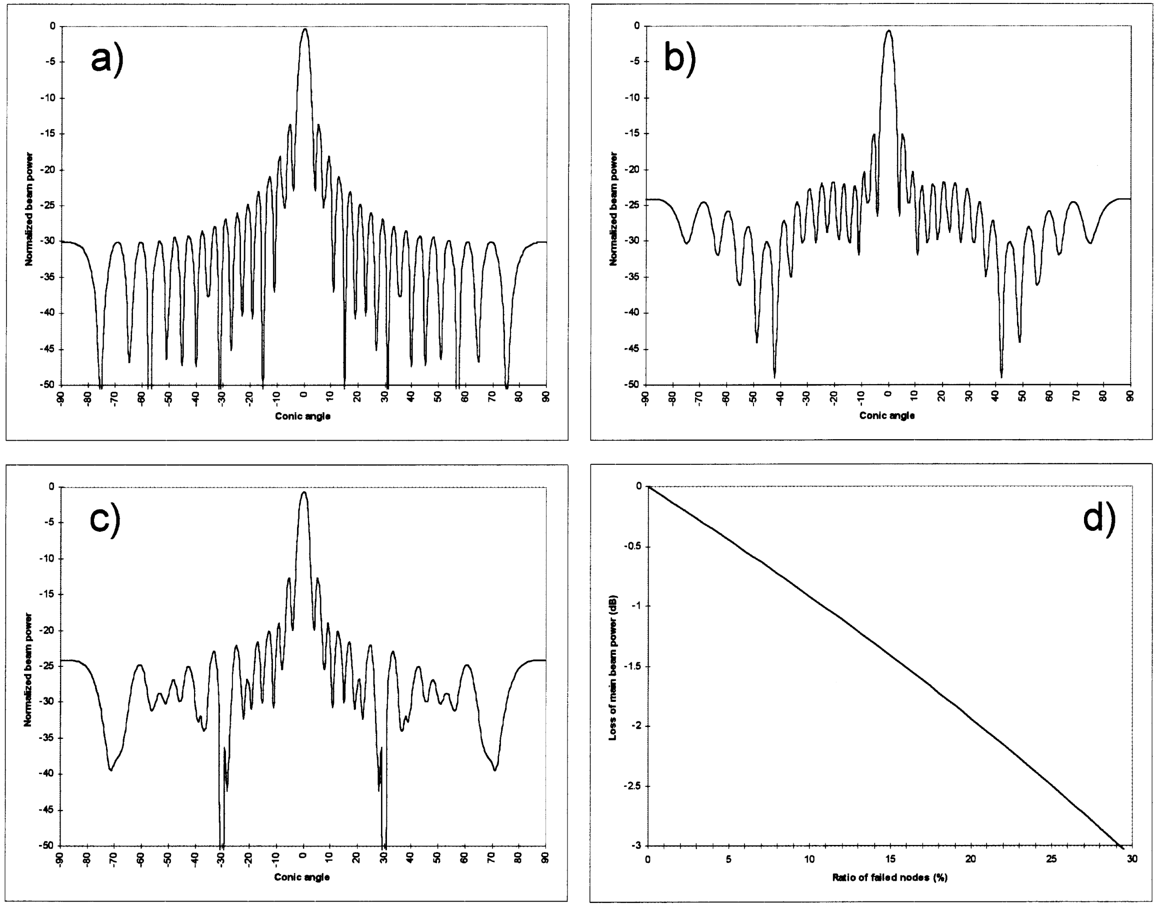

Fig. 2. Normalized beam power patterns for a 32-node array (a), (b), and (c) for the cases of no failed nodes, node 4 failed, and node 14 failed, respectively. (d) Loss in main beam power as a function of the percentages of nodes failed.

three 32-node arrays with half-wavelength spacing. Pattern (a) is that of an array with no failed nodes. Patterns (b) and (c) are those of an array with a single failed node (nodes 4 and 14 respectively). The side lobe pattern is strongly dependent on the location of the failed nodes. However, the loss in the main beam power is independent of their location and depends only on the number of failed nodes. Figure 2 (d) shows the loss in main beam power vs the percentage of nodes that have failed. From this figure it is apparent that a substantial number of nodes may fail before the main beam power is significantly degraded.

In order for the network to maintain connectivity in the case of multiple failures it is necessary that the number of adjacent failed nodes not exceed the number that may be bypassed. The probability that the network will fail if the number of adjacent failed nodes exceeds the number that may be bypassed may be determined from simple combinatorial logic. For a network of $N$ nodes of which $b$ adjacent nodes may be bypassed the probability of network failure, $P(F)$, due to any $b+1$ adjacent nodes failing is

$$P(F) = 1-(1-p^{b+1})^{N-(b+1)},$$

where $p$ is the probability of any one node failing. For small $p$

$$P(F) \simeq [N-(b+1)]p^{b+1}.$$

Figure 3 shows the probability of network failure vs the number of adjacent nodes which may be bypassed for varying numbers of failed nodes in a network of 112 nodes. The number 112 is chosen because a typical array configuration might consist of two 64-node subarrays and one 32-node subarray nested in a common array of 112 nodes. For comparison, the probability that 14 of the 112 nodes in the common array fail is the same as if 4 nodes fail in a 32-node subarray. Figures 2 and 3 suggest that it will be necessary to bypass three or more adjacent failed nodes if the network is to maintain connectivity while the array is still taking useable data.

The number of adjacent failed nodes that may be bypassed depends on the type of bypass that is used. Optical bypass switches are of two types — passive and active. The passive switch operates by shunting a part of the optical signal (typically less than $-10$ dB) around the node and mixing it with the outgoing signal. In the event that the node fails this shunted signal is then the only one seen by the next node on the network. Passive bypasses have the advantages that, with no active components, they are extremely reliable and consume no electrical power. The principal disadvantage is that the attenuation from adjacent failed nodes accumulates, limiting the number of nodes that may be bypassed to one or two, depending on the dynamic range of the receiver in the downstream node. Active bypasses rely on mechanical switching or effects such as the electro- or acousto-optic effect and are capable of bypassing a relatively large number of failed nodes. However, currently available devices tend to be less reliable, large, and either power hungry or expensive or both. Micromachine technology offers the promise of addressing all of these problems and low-cost reliable 2 × 2 bypass switches are currently under development. These switches will be capable of bypassing up to five failed nodes with power consumption

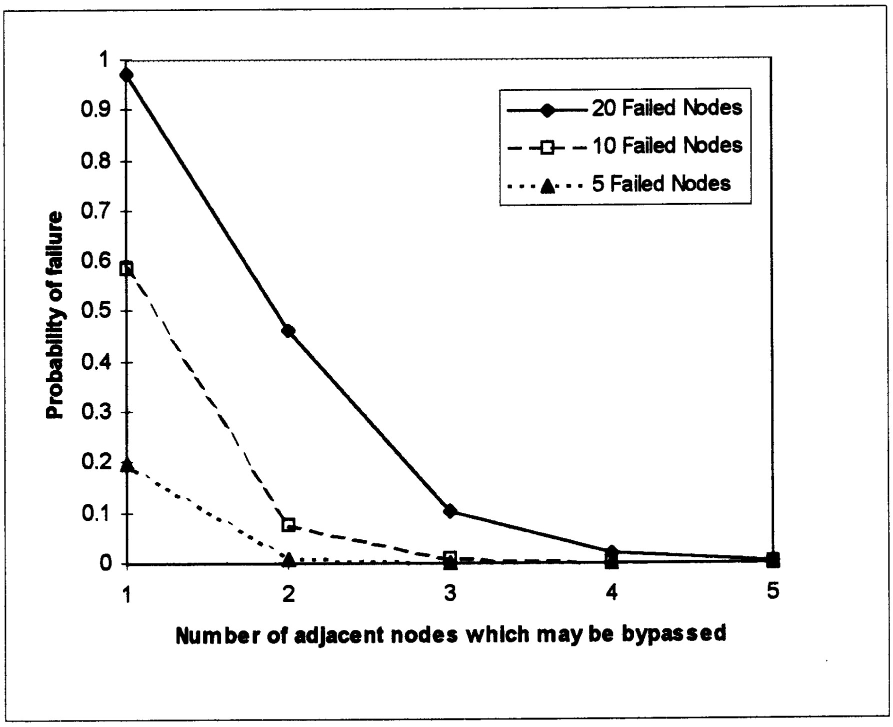

Fig. 3. Probability of network failure in a 112-node array as a function of the number of adjacent failed nodes which may be bypassed for the cases of 1, 5, 10, and 20 randomly failed nodes.

less than 300 nW at 3 V. The packaged version of these switches will be smaller than 1 cm$^3$. The $2 \times 2$ switch architecture has the added advantage that in the *bypass* state the node's output is connected directly to its input, providing a self-test capability [2].

The problem of degraded sidelobes is more complex. A node failure is equivalent to setting the weighting function for the corresponding node to zero, which represents a very poor choice of filtering function. The problem has been considered in the case of phased-array antennas [3]. In this case sidelobe suppression may be accomplished by varying the weighting function, phase, frequency, or element position. For autonomous arrays only the first two may be varied. The difficulty with adjusting the weighting function is that any function other than a uniform one will result in decreased sensitivity in the main beam. For an autonomous array this may be an acceptable tradeoff in exchange for greatly improved sidelobe suppression and simplified processing. As an example, for the case of Fig. 2(b) in which a node near the end has failed (node 4), a large improvement in beam power pattern may be obtained by simply setting the weighting functions of nodes 1–3 to zero. The improvement in beam power pattern can be seen in Fig. 4. The pattern for an array with no failed nodes is superimposed (solid line). The additional decrease in main power is 0.44 dB. This solution has the added advantage of decreasing the overall processing load on

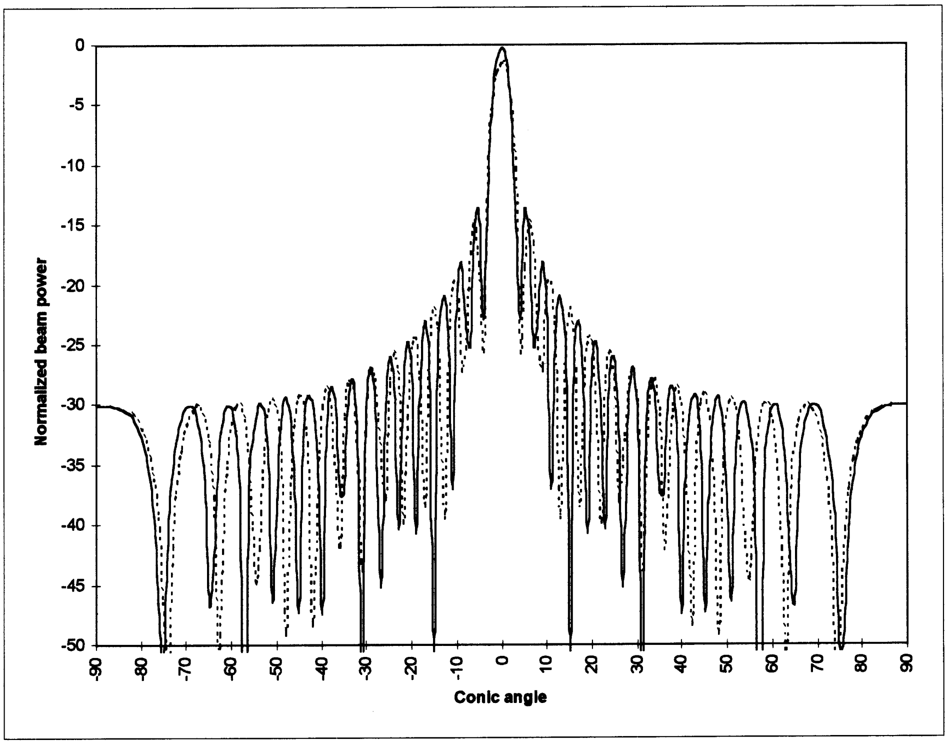

Fig. 4. Beam power pattern for a 32-node array in which node 4 has failed and the weighting functions for nodes 1–3 have been set to zero (dashed line). The solid line represents the pattern for a 32-node array with no failed nodes.

the surviving nodes. For failed nodes nearer the center of the array this relatively simple solution is not an option. In this case the feasibility of using more complex weighting functions and phase variations needs to be studied.

## 4. Parallel and distributed processing architecture

In the original system concept the acoustic array was designed to pass data via the fiber-optic cable to a stand-alone data collector/processor. This processor represented a major single-point-of-failure, performance bottleneck, and cost driver. System reliability can be greatly enhanced by replacing the stand-alone processor with a processing architecture in which each node of the network represents a processing element of a parallel processor, essentially turning the array itself into a distributed processing machine. This approach offers the potential for greatly reduced cost with increased system performance, dependability, and versatility. Furthermore, by using the spare processing capacity in the processors used to implement the network protocol together with the high data rate offered by fiber optics, these improvements can be achieved at essentially no increase to the per-node cost of the array. In this section we present parallel algorithms and performance models for the decomposition and mapping of frequency-domain beamforming algorithms to a number of array topologies.

For the purpose of detection and location of surface and underwater objects, the execution of conventional and adaptive beamforming programs are the primary function of these autonomous acoustic sonar arrays. The initial algorithms being emphasized in this work involve sequential and parallel processing techniques in the frequency domain for conventional beamforming. The fundamental computational component of these algorithms is the radix-2 fast Fourier transform (FFT).

### *4.1. Sequential FFT beamforming*

As shown in the algorithm depicted by Fig. 5, once the data values in the current sample set have been collected from the array nodes, the first step of the beamformer is to perform the FFT and windowing functions for each node's data samples. Next, a loop is entered which iterates once for each of the steering directions. For each angle, the transformed data from each node is multiplied by the node-dependent value for the steering factor, the data is summed, and the results are inverse-transformed and the magnitude calculated which gives the beamform output for the current steering direction. The set of beamform outputs for all steering angles is collected at the end, and the entire process is repeated for successive iterations, each with a new set of input samples from the acoustic transducers.

### *4.2. Parallel FFT beamforming*

A wide array of parallel algorithms can be constructed from the baseline sequential algorithm. Each of the loops in the sequential algorithm can be decomposed into partitions which are mapped to processors in the distributed parallel sonar array. The effectiveness and efficiency of this parallelization is of course dependent upon the system architecture and network topology to be targeted. For example, Fig. 6 illustrates one method employed for parallel FFT beamforming mapped to a uni-directional linear array. The most downstream node in the array must do the most work, since the algorithm requires a summation of the data from all nodes. Since communication is only one-way, the only node capable of receiving data from all nodes is the most downstream node. This condition severely limits the degree of parallelism possible, and the array is not evenly balanced in terms of computational load.

An enhanced version, shown in Fig. 7, is achieved by removing the steering direction loop from the purview of the first node and distributing most of the computations in the loop across all the nodes. In the process, rather than sending column after column to the front-end node (i.e. the node responsible for the final collection of beamform results and subsequent transmission of these results via the array uplink), each node

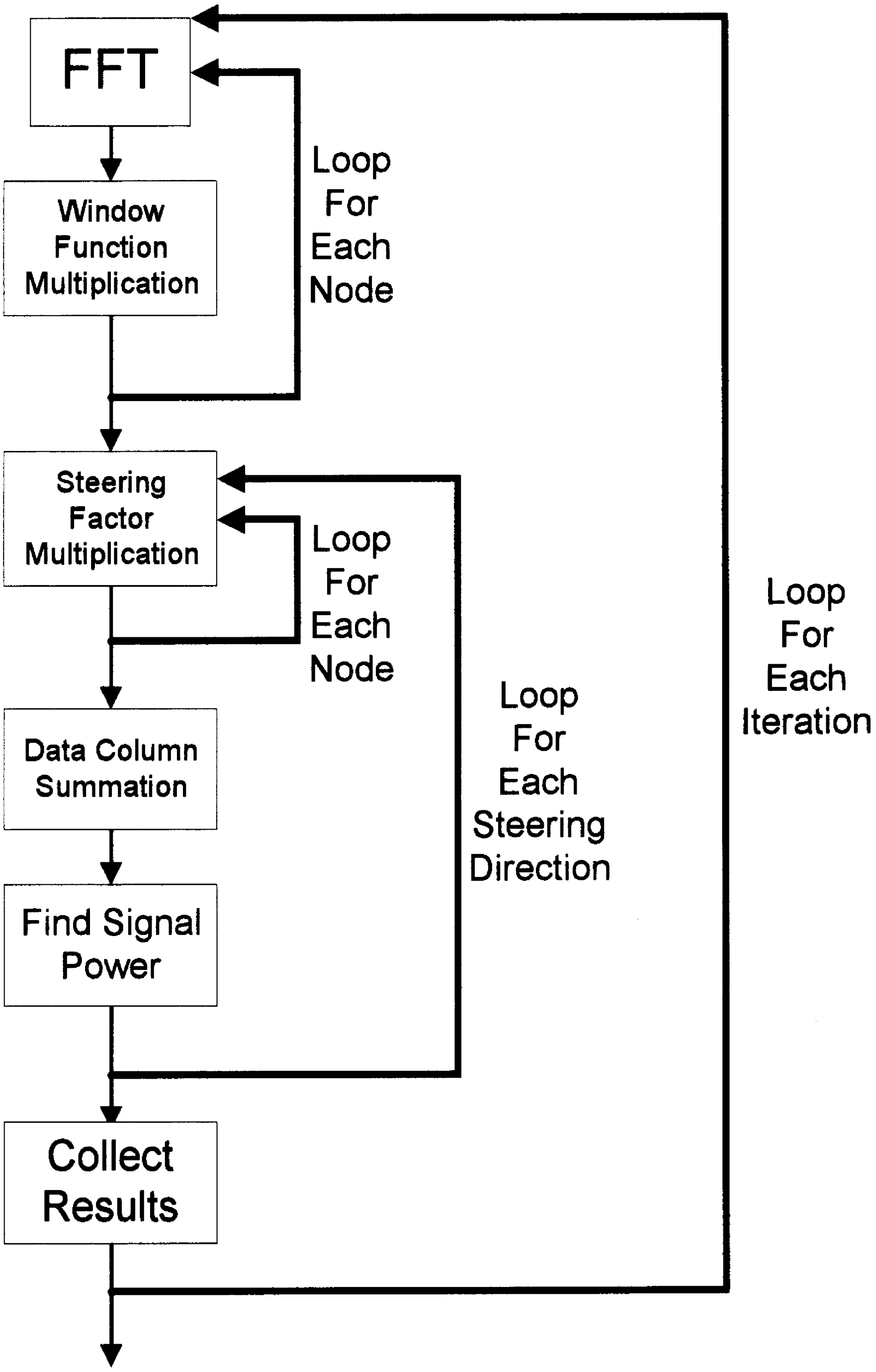

Fig. 5. Sequential FFT beamforming.

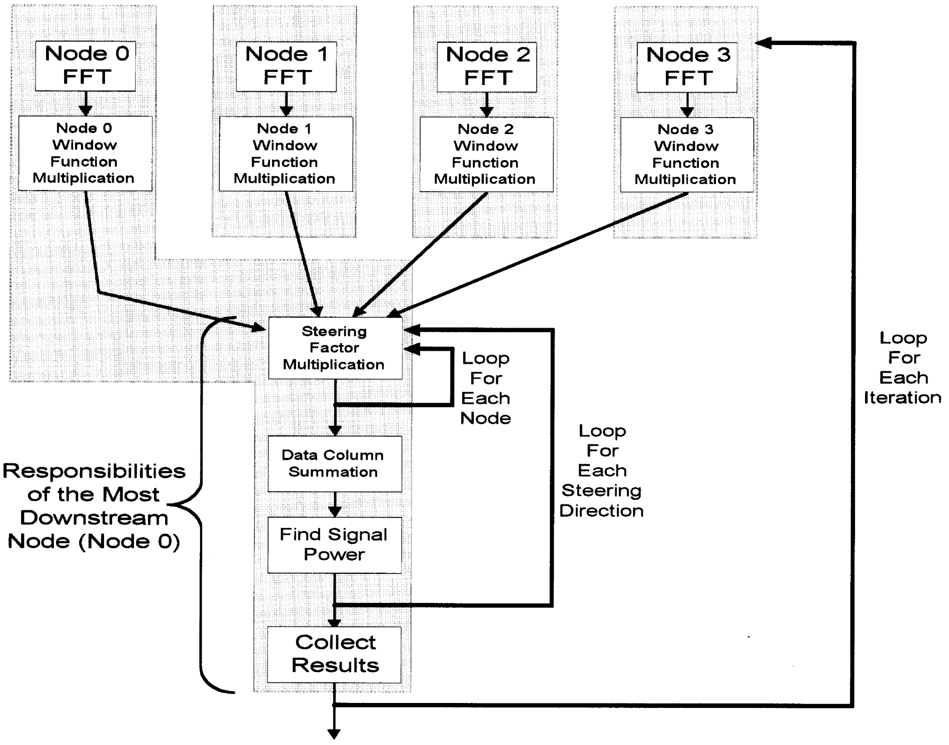
Node 0 FFT
Node 1 FFT
Node 2 FFT
Node 3 FFT
Node 0 Window Function Multiplication
Node 1 Window Function Multiplication
Node 2 Window Function Multiplication
Node 3 Window Function Multiplication
Steering Factor Multiplication
Loop For Each Node
Data Column Summation
Find Signal Power
Collect Results
Loop For Each Steering Direction
Loop For Each Iteration
Responsibilities of the Most Downstream Node (Node 0)

Fig. 6. Basic parallel beamformer for uni-directional linear arrays (a 4-node array is depicted for simplicity).

receives a column from upstream, adds it column to it, and sends the resulting column downstream.

With sonar array architectures which support more sophisticated network topologies, such as the bi-directional linear array or the ring, more advanced parallel beamforming algorithms can be exploited. For example, as illustrated in Fig. 8, data can be exchanged between any pair of nodes in the array network, so that the use of a floating front-end processor may offer concurrent execution of multiple iterations of the beamforming algorithm. Since iterations are at the highest level of the algorithm (i.e. the outermost loop), and communication between iterations is not required, this approach offers the benefit of coarse-grain parallelism which supports systems with limited interprocessor communication bandwidth.

Rather than forcing all data to be sent to the same front-end processor every time, the data can be sent to a different node each iteration, and the many iterations can be computed in an overlapping or pipelining fashion. For instance, the nodes can collect another sample set from the acoustic transceivers and begin the second iteration before the first iteration is completed. At the beginning of the second iteration, the node doing the front-end work of the first iteration temporarily stops its work long

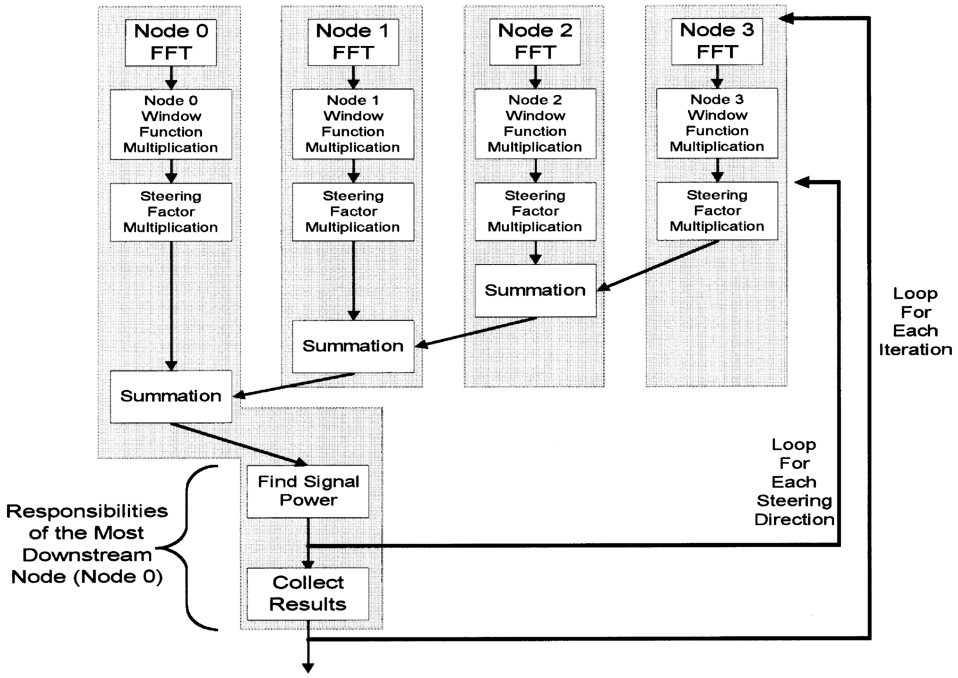
Node 0 FFT
Node 1 FFT
Node 2 FFT
Node 3 FFT
Node 0 Window Function Multiplication
Node 1 Window Function Multiplication
Node 2 Window Function Multiplication
Node 3 Window Function Multiplication
Steering Factor Multiplication
Steering Factor Multiplication
Steering Factor Multiplication
Steering Factor Multiplication
Summation
Summation
Summation
Loop For Each Iteration
Loop For Each Steering Direction
Find Signal Power
Collect Results
Responsibilities of the Most Downstream Node (Node 0)

Fig. 7. Enhanced parallel beamformer for uni-directional linear arrays.

enough to transform its portion of the data from the second iteration, multiply by the windowing factor, and send it off to the second-iteration front-end processor (i.e. a processor located in a neighboring node). Once the data has been sent to the new iteration front-end processor, the node resumes front-end work for the previous iteration as before. The process continues for additional iterations. As the iterations progress, an increasing number of nodes become involved in front-end work for respective iterations, all in different stages of completion.

Whereas the beamformer depicted in Fig. 8 exploits the parallelism present between iterations, parallelism within each iteration can also be addressed. By distributing the processing of the many steering directions per iteration across the processors in the array, a medium-grain parallel algorithm is obtained. For example, for a 30-node array with 90 steering directions per iteration, processor for each node would be responsible for computing three of the steering directions per iteration. As shown in Fig. 9, this algorithm uses a fixed front-end processor, but the only additional processing required of this processor is the simple collection of results.

Using a testbed consisting of workstations connected by networks including ATM, SCI, and Ethernet, experiments with these and other new parallel beamfomers have shown an execution-time speedup which increases in a near-linear fashion with respect to an increase in processors. For instance, Fig. 10 illustrates the performance charac-

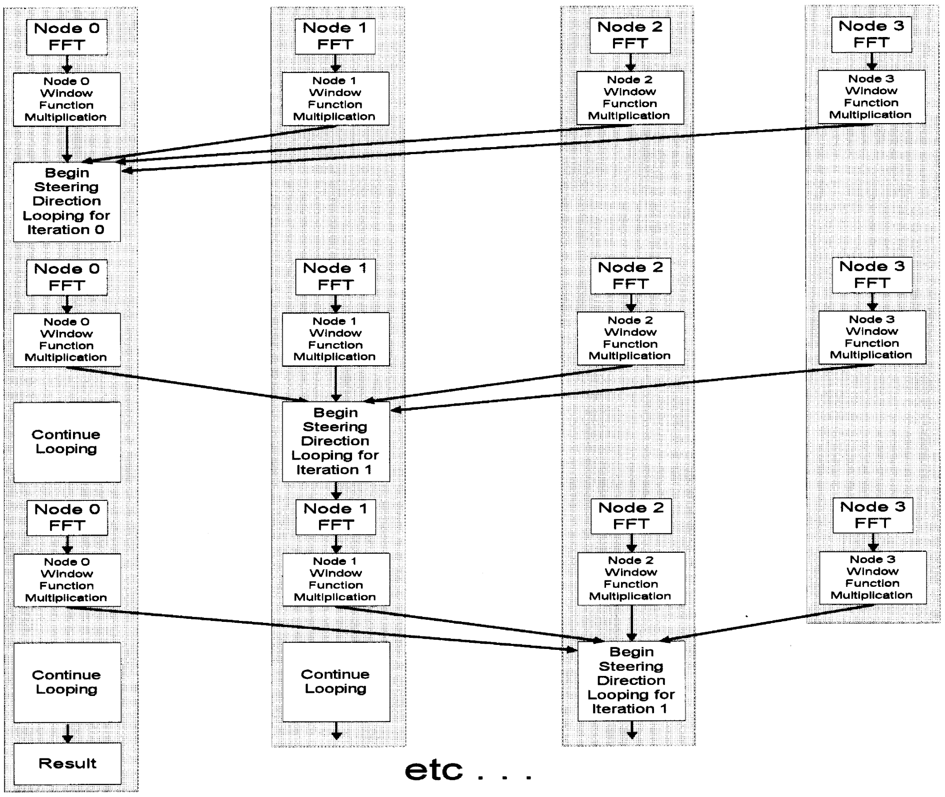
Node 0 FFT
Node 0 Window Function Multiplication
Node 1 FFT
Node 1 Window Function Multiplication
Node 2 FFT
Node 2 Window Function Multiplication
Node 3 FFT
Node 3 Window Function Multiplication
Begin Steering Direction Looping for Iteration 0
Node 0 FFT
Node 0 Window Function Multiplication
Node 1 FFT
Node 1 Window Function Multiplication
Node 2 FFT
Node 2 Window Function Multiplication
Node 3 FFT
Node 3 Window Function Multiplication
Continue Looping
Begin Steering Direction Looping for Iteration 1
Node 0 FFT
Node 0 Window Function Multiplication
Node 1 FFT
Node 1 Window Function Multiplication
Node 2 FFT
Node 2 Window Function Multiplication
Node 3 FFT
Node 3 Window Function Multiplication
Continue Looping
Continue Looping
Begin Steering Direction Looping for Iteration 1
Result
etc . . .

Fig. 8. Coarse-grain parallel beamformer for bi-directional and ring arrays.

teristics of the coarse- and medium-grain parallel beamformers on a cluster of workstations connected by OC-3c ATM, where speedup is defined as the ratio of the sequential execution time to the parallel execution time. A high degree of parallel efficiency makes it possible for a distributed parallel sonar array to address trends which require the implementation of high-element-count sonar arrays and lead to a corresponding increase in data rate and the associated signal processing. Furthermore, increased signal processing efficiency also makes it possible to reduce array power requirements, since the parallel versions of some beamformers are sufficiently fast as to permit real-time deadlines to be met with a surplus in time. This surplus can be exploited in low-power standby mode or by reducing the clock rate of the processors to reduce array power consumption of the battery-powered sonar array system. In addition, the increased processing efficiency often results in decreased data rates between processors. This decrease in data rate decreases the power requirements of the fiber-optic links further increasing battery lifetime.

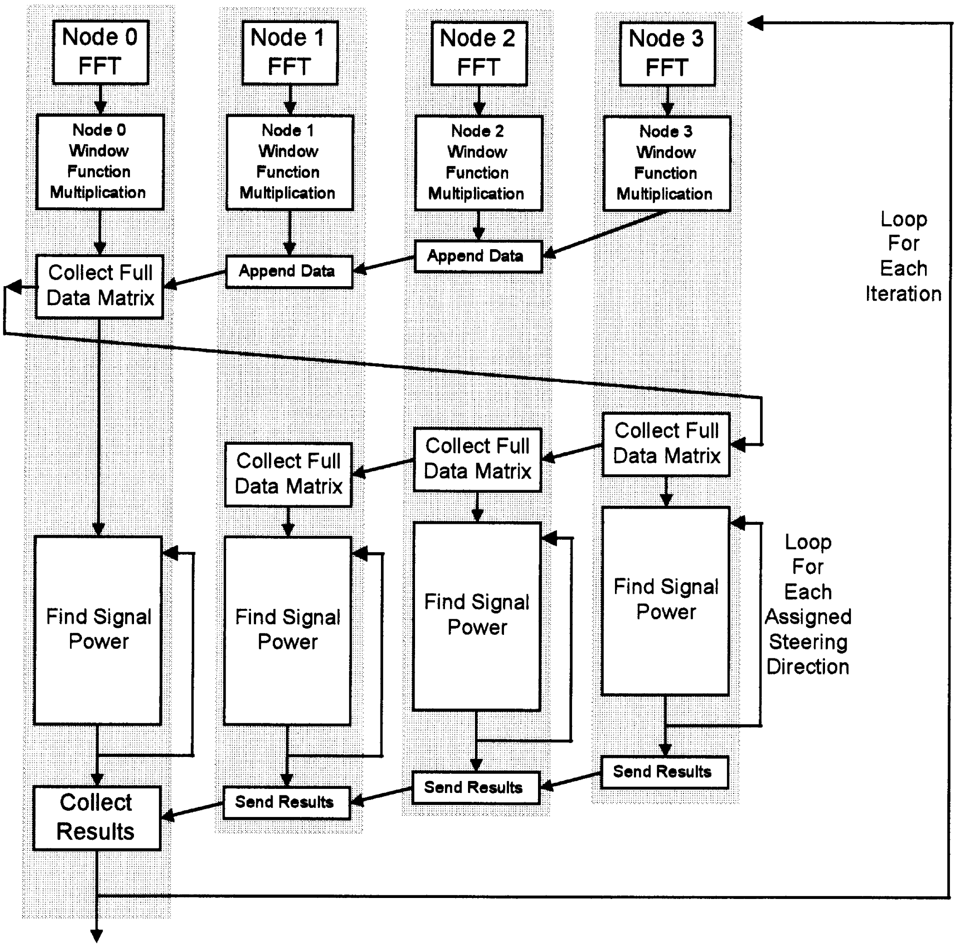

Fig. 9. Medium-grain parallel beamformer for bi-directional and ring arrays.

## 5. Conclusions

In this paper we have considered the problem of fault tolerance in autonomous disposable acoustic arrays. The principal source of failures over relatively short mission times is battery run-down resulting in node outage. In order to maintain network continuity in the presence of such failures it is essential to be able to bypass several adjacent failed nodes. The problem of degraded sidelobe pattern under these circumstances is amenable to relatively simple solutions for the case of failures near the ends of the array but failures near the center are more intractable and are the subject of future study. New distributed parallel beamforming algorithms and architectures provide increased fault tolerance in terms of the loss of critical processing elements, near-linear speedup and scalability, and reduced power requirements.

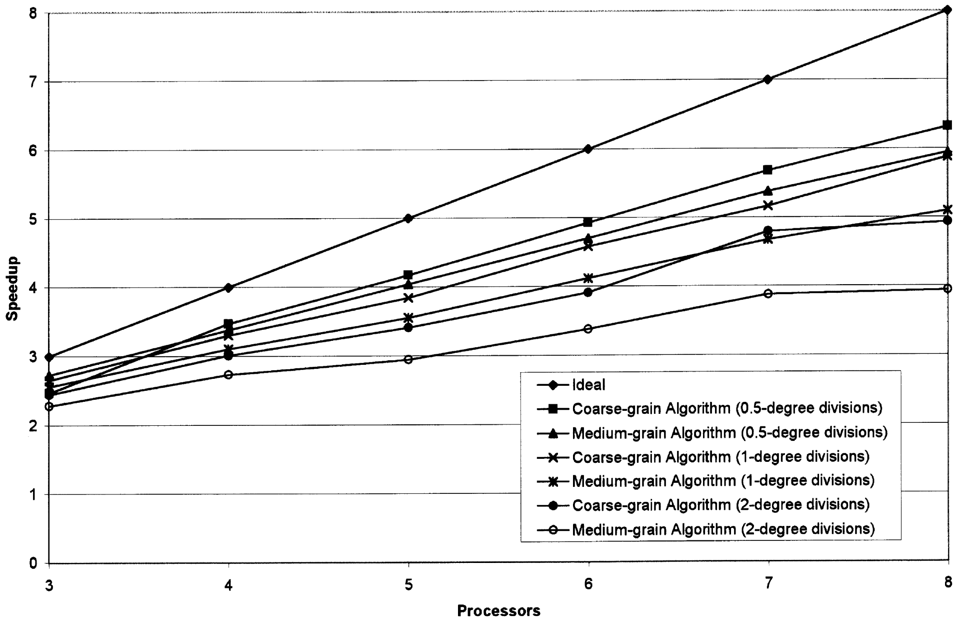

Fig. 10. Performance of coarse- and medium-grain parallel beamformers on a cluster of eight SPARCstation-20/85 workstations connected by OC-3c ATM (155-Mbps), for steering directions from −90 to 90° in increments of 0.5, 1, or 2°.

## Acknowledgements

The authors wish to express their thanks to Drs John Tague and Don Davison, and CDR Mitch Shipley of the Office of Naval Research for their continued support. We also thank Jeff Markwell and Ryan Fogarty, research assistants at the University of Florida, for their assistance in this effort and Dr Afshin Daryoush of Drexel University for his many useful suggestions. This work was supported by grants N00014-96-1-0569 and N00014-97-1-0117 from the Office of Naval Research.

## References

[1] A. George, W. Rosen, J. Markwell, L. Hopwood, and R. Fogarty, "Cluster computing architectures and algorithms for passive sonar arrays," *Proceedings of the Cluster Computing Conference*, Atlanta, Georgia, March 1997.

[2] J. Greene, R. Crotts, C. Kozikowski, M. Miller and K. Murphy, "Battery-powered optical fiber bypass switch for military and private sector communications networks," *Technical Progress Report on Navy Contract N*00421-96-*C*-1127, March, 1997.

[3] A. Daryoush and B. Choe, "Optically reconfigured active phased array antennas," *Microwave and Optical Technology Letters*, Vol. 1, No. 9, pp. 344–348, November 1988.